

**LAPORAN PRAKTIKUM**  
**AGORITMA PEMROGRAMAN**  
**“CLASS SCANNER DAN JENIS-JENIS OPERATOR”**

**DISUSUN OLEH:**  
**ARYAHYAHULFIKRA**  
**2511532026**

**DOSEN PENGAMPU:**  
**Dr. WAHYUDI, S.T, M.T**  
**ASISTEN PRAKTIKUM:**  
**RAHMAD DWIRIZKI**



**DEPARTEMEN INFORMATIKA**  
**FAKULTAS TEKNOLOGI INFORMASI**  
**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS ANDALAS**  
**2025**

## **KATA PENGANTAR**

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya, Laporan praktikum ini dapat diselesaikan dengan baik. Laporan praktikum ini disusun sebagai salah satu tugas dalam rangka memenuhi tugas yang diberikan oleh dosen dan juga asisten praktikum.

Padang, 26 September 2025

Aryahiyahul fikra

## DAFTAR ISI

<b>KATA PENGANTAR .....</b>	<b>i</b>
<b>BAB I .....</b>	<b>1</b>
<b>PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Tujuan.....	1
1.3 Manfaat .....	2
<b>BAB II PEMBAHASAN.....</b>	<b>3</b>
2.1 Pengertian .....	3
2.2 Pengertian Operand dan Operator .....	3
2.3 Pemograman Pekan 3 .....	4
2.3.1 Pemograman Assignment.....	4
2.3.2 Pemograman operasi logika.....	4
2.3.3 Pemograman relasional .....	5
<b>BAB III .....</b>	<b>6</b>
<b>PENUTUP.....</b>	<b>6</b>
3.1 Kesimpulan .....	6
3.2 Saran .....	6
<b>DAFTAR PUSTAKA.....</b>	<b>8</b>

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Perkembangan teknologi informasi yang pesat dalam beberapa dekade terakhir telah memberikan dampak besar terhadap berbagai aspek kehidupan manusia, termasuk di bidang pendidikan, industri, dan komunikasi. Salah satu pilar utama dalam perkembangan teknologi tersebut adalah **pemrograman komputer**. Di antara banyak bahasa pemrograman yang ada, **Java** merupakan salah satu bahasa yang sangat populer dan banyak digunakan di berbagai belahan dunia.

Java adalah bahasa pemrograman tingkat tinggi yang bersifat **object-oriented**, dikembangkan oleh Sun Microsystems pada tahun 1995, dan kini dikelola oleh Oracle. Keunggulan utama Java terletak pada kemampuannya untuk berjalan di berbagai platform (platform-independent) berkat konsep **Write Once, Run Anywhere (WORA)** yang dimilikinya. Hal ini memungkinkan program yang dibuat dalam Java untuk dijalankan di berbagai sistem operasi tanpa perlu diubah ulang.

Selain itu, Java memiliki sintaks yang relatif mudah dipahami bagi pemula, dokumentasi yang lengkap, dan komunitas pengguna yang besar, sehingga sangat mendukung proses belajar dan pengembangan perangkat lunak. Java juga menjadi bahasa utama dalam pengembangan berbagai aplikasi, seperti aplikasi desktop, mobile (Android), web, hingga sistem enterprise berskala besar.

Dengan melihat pentingnya Java dalam dunia pemrograman dan aplikasinya yang luas, sangatlah relevan untuk mempelajari dasar-dasar bahasa pemrograman Java, baik dari segi teori maupun praktik. Oleh karena itu, laporan ini disusun untuk memberikan pemahaman yang lebih mendalam tentang bahasa Java, struktur dasarnya, serta penerapan konsep pemrograman berorientasi objek dalam bahasa tersebut.

### **1.2 Tujuan**

1. Membantu mahasiswa memahami konsep dasar pemrograman Java melalui penerapan langsung.
2. Melatih kemampuan menulis, mengompilasi, dan mengeksekusi program dengan

mengikuti aturan sintaksis Java.

3. Meningkatkan keterampilan dalam memecahkan masalah (problem solving) dengan pendekatan algoritmik
4. Membiasakan mahasiswa bekerja sistematis dalam menyusun laporan yang memuat analisis hasil praktikum
5. Menanamkan sikap teliti, disiplin, serta tanggung jawab dalam melaksanakan kegiatan laboratorium.

### **1.3 Manfaat**

1. Mahasiswa dapat memahami konsep dasar pemrograman secara praktis, bukan hanya teori.
2. Melatih keterampilan logika dan analisis dalam menyusun serta mengeksekusi kode.
3. Memberikan pengalaman langsung dalam menggunakan bahasa Java sebagai salah satu bahasa pemrograman populer.
4. Menjadi bekal awal untuk memahami materi pemrograman yang lebih kompleks pada praktikum berikutnya.
5. Membiasakan mahasiswa dengan proses debugging dan pemecahan masalah dalam pemrograman.

## **BAB II**

### **PEMBAHASAN**

#### **2.1 Pengertian**

Scanner dalam Java adalah sebuah kelas dari paket `java.util` yang digunakan untuk membaca input dari berbagai sumber, seperti konsol (keyboard), berkas, atau bahkan string, serta untuk mengurai tipe data primitif (seperti `int`, `double`) dan `String` dari input tersebut. Kelas ini memecah input menjadi token berdasarkan spasi secara default dan menyediakan berbagai metode untuk membaca data sesuai kebutuhan, seperti `nextInt()` untuk membaca bilangan bulat atau `nextLine()` untuk membaca satu baris teks.

**Kemudahan Penggunaan:** Scanner Class menyediakan antarmuka yang mudah digunakan untuk mengambil input dari pengguna. Dengan metode-metodenya yang intuitif, seperti `nextInt()`, `nextDouble()`, dan `nextLine()`, pengembang dapat dengan mudah mengambil berbagai jenis input tanpa harus menulis kode yang rumit.

**Fleksibilitas:** Scanner Class dapat digunakan untuk mengambil input dari berbagai sumber, termasuk keyboard (`System.in`), file teks, atau bahkan string. Ini memberikan fleksibilitas yang besar dalam pengembangan aplikasi, terutama ketika perlu untuk mengambil input dari sumber yang berbeda.

**Dukungan untuk Berbagai Tipe Data:** Scanner Class mendukung berbagai jenis tipe data, seperti `int`, `double`, `String`, dan lain-lain. Ini memudahkan untuk mengambil input dalam berbagai format dan mengonversinya ke tipe data yang sesuai.

#### **2.2 Pengertian Operand dan Operator**

**Operand** adalah nilai asal yang dipakai dalam sebuah proses operasi. Sedangkan **Operator** adalah instruksi yang diberikan untuk mendapatkan hasil dari proses tersebut.

## 2.3 Pemograman Pekan 3

### 2.3.1 Pemograman Assigment

```
1 package pekan3;
2 import java.util.Scanner;
3
4 public class OperasiAssigment {
5
6     public static void main(String[] args) {
7         int A1;
8         int A2;
9         int hasil;
10
11         Scanner keyboard = new Scanner(System.in);
12         System.out.println("Input angka-1:");
13         A1 = keyboard.nextInt();
14         System.out.println("Input angka-2:");
15         A2 = keyboard.nextInt();
16         keyboard.close();
17         System.out.println("Assigment Penjumlahan");
18         A1 += A2; // Penambahan sekarang A1 = 15
19         System.out.println("Penambahan : " + A1);
20         System.out.println("Assigment pengurangan");
21         A1 -= A2; // Pengurangan sekarang A1 = 10
22         System.out.println("Pengurangan : " + A1);
23         System.out.println("Assigment perkalian");
24         A1 *= A2; // Perkalian sekarang A1 = 50
25         System.out.println("Perkalian : " + A1);
26         System.out.println("Assigment hasil bagi");
27         A1 /= A2; // Hasil bagi sekarang A1 = 10
28         System.out.println("Pembagian : " + A1);
29         System.out.println("Assigment sisa bagi");
30         A1 %= A2; // Sisa bagi sekarang A1 = 0
31         System.out.println("Sisa bagi : " + A1);
32     }
33 }
34
```

Gambar 2.3.1

Pada pemograman ini kita belajar bahwa pemograman assigment sama seperti pemograman aritmatika akan tetapinyang membedakan adalah fungsi dari pemograman ini yang lebih berfokus pada penilaian assigment.

### 2.3.2. Pemograman operasi logika

```
1 package pekan3;
2 import java.util.Scanner;
3
4 public class OperasiLogika {
5
6
7     public static void main(String[] args) {
8         boolean A1;
9         boolean A2;
10        boolean c;
11        Scanner keyboard = new Scanner(System.in);
12        System.out.print("Input nilai boolean-1 (True / False): ");
13        A1 = keyboard.nextBoolean(); //Input 10
14        System.out.print("Input nilai boolean-2 (True / False): ");
15        A2 = keyboard.nextBoolean();
16        keyboard.close();
17        System.out.println("A1 = " + A1);
18        System.out.println("A2 = " + A2);
19        System.out.println("Konjugnsi ");
20        c = A1&&A2;
21        System.out.println("true and false" + c);
22        System.out.println("Disjungsi");
23        c = A1|A2;
24        System.out.println("true or false" + c);
25        System.out.println("Nagasi");
26        c = !A1;
27        System.out.println("Bukan true " + c);
28        //
29
30    }
31 }
32
33 }
34
```

Pada pemograman ini kita menggunakan boolean yang berfungsi untuk menyimpan nilai sementara yang berguna untuk mengoperasikan pemograman ini untuk menentukan benar atau salahnya sesuatu.

### 2.3.3. Pemograman relasional

```
1 package pekan3;
2
3 import java.util.Scanner;
4
5 public class OperasiRelasional {
6
7     public static void main(String[] args) {
8         int A1;
9         int A2;
10        boolean hasil;
11
12        Scanner keyboard = new Scanner(System.in);
13        System.out.println("Input angka-1:");
14        A1 = keyboard.nextInt();
15        System.out.println("Input angka-2:");
16        A2 = keyboard.nextInt();
17        keyboard.close();
18        hasil = A1 > A2; // Apakah A1 lebih besar A2?
19        System.out.println("A1 > A2 = " + hasil);
20        hasil = A1 < A2; // Apakah A1 lebih kecil A2?
21        System.out.println("A1 < A2 = " + hasil);
22        hasil = A1 >= A2; // Apakah A1 lebih besar sama dengan A2?
23        System.out.println("A1 >= A2 = " + hasil);
24        hasil = A1 <= A2; // Apakah A1 lebih kecil sama dengan A2?
25        System.out.println("A1 <= A2 = " + hasil);
26        hasil = A1 == A2; // Apakah A1 sama dengan A2?
27        System.out.println("A1 == A2 = " + hasil);
28        hasil = A1 != A2; // Apakah A1 tidak sama dengan A2?
29        System.out.println("A1 != A2 = " + hasil);
30        //
31    }
32 }
33
34 }
35
```

Gambar 2.3.4

Pada pemograman kali ini kita bisa melihat bahwa pemograman diatas menjelaskan tentang cara mencari antara 2 bilangan mana yang lebih besar,mana yang lebih kecil,mana yang sama dengan dan mana yang tidak sama dengan.



## **BAB III**

### **PENUTUP**

#### **3.1 Kesimpulan**

Dengan memahami dan menguasai penggunaan **Scanner**, **operand**, dan **operator**, seorang pemrogram dapat membuat program Java yang **interaktif**, **logis**, dan **berdaya guna**. Scanner memungkinkan komunikasi dua arah antara program dan pengguna, sementara operand dan operator adalah komponen kunci dalam membentuk **ekspresi**, **logika**, dan **alur program**.

Ketiganya merupakan **konsep dasar** yang saling terkait dan tidak dapat dipisahkan dalam dunia pemrograman, terutama dalam menyusun program-program berskala kecil hingga menengah. Oleh karena itu, pemahaman terhadap ketiga konsep ini harus dikuasai secara menyeluruh sebelum mempelajari materi Java yang lebih kompleks, seperti **struktur kontrol**, **array**, **class & object**, dan lain sebagainya.

#### **3.2 Saran**

Berdasarkan pembahasan mengenai kelas Scanner, operand, dan operator dalam bahasa pemrograman Java, terdapat beberapa hal yang dapat disarankan untuk proses pembelajaran dan pengembangan kemampuan pemrograman, khususnya bagi pemula:

##### **1. Pahami Dasar-Dasarnya Secara Konseptual dan Praktis**

Pemahaman teori mengenai fungsi Scanner, jenis-jenis operator, dan cara kerja operand sangat penting sebagai landasan awal. Namun, selain memahami konsep secara teoritis, sangat disarankan untuk mempraktikkannya langsung melalui latihan coding sederhana. Semakin sering berlatih, semakin mudah mengenali pola dan logika program.

##### **2. Perbanyak Latihan Soal dan Proyek Mini**

Untuk memperkuat pemahaman, mahasiswa atau pembelajar disarankan untuk

mengerjakan berbagai latihan soal, seperti membuat program kalkulator sederhana, program login, atau sistem input data. Proyek mini seperti ini sangat efektif dalam mengasah keterampilan teknis, terutama dalam penggunaan Scanner untuk input dan operator untuk pengolahan data.

### 3. Waspada Error Umum

Dalam menggunakan Scanner, perhatikan perpindahan antar metode input, terutama ketika mencampurkan `nextInt()` dan `nextLine()`. Banyak pemula mengalami error karena tidak mengetahui bahwa `nextInt()` tidak membaca karakter newline (`\n`), sehingga baris input berikutnya dapat terlewat atau kosong. Saran teknis: selalu tambahkan `input.nextLine()`; setelah membaca angka jika ingin melanjutkan ke input teks.

### 4. Gunakan Komentar dan Format Kode yang Rapi

Meskipun masih dalam tahap belajar, pembiasaan menulis kode dengan struktur yang rapi dan menambahkan komentar sangat membantu dalam membaca, memahami, dan memperbaiki program di kemudian hari.

### 5. Manfaatkan Sumber Belajar Tambahan

Disarankan untuk memperluas pemahaman dengan membaca dokumentasi resmi dari Oracle dan menggunakan platform belajar daring seperti W3Schools, GeeksforGeeks, atau JavaPoint. Menonton tutorial interaktif di YouTube juga bisa menjadi metode belajar yang menarik

## **DAFTAR PUSTAKA**

<https://www.duniaikom.com/tutorial-belajar-java-jenis-jenis-operator-dalam-bahasa-java/>

<https://www.gramedia.com/literasi/operator-adalah/>

<https://www.duniaikom.com/tutorial-belajar-java-jenis-jenis-operator-dalam-bahasa-java/>